

Reuse Framework Study

Team: Gail McConaughy, GSFC
Mark Nestler, GST
David Isaac, Business Performance Systems/GST
Nadine Alameh, GST

SEEDS Third Public Workshop

Annapolis, MD

March 18-20, 2003

□ Background

- Study Findings and Recommendations
- Process Guidelines

□ Workshop plans

- Developing Draft Charters of Reuse Working Groups
 - Purpose
 - Scope
 - Membership
 - Goals & Metrics
 - Initial Steps
- Gathering Community Opinion on SEEDS Reuse Policy
 - Introduction
 - Purpose of Session
 - Community Input Regarding Current NASA Policies
 - Community Input Regarding External Organizations Policies
 - Initial Community Policy Recommendations
 - For Mission-Critical Community
 - For Mission-Success Community

Background

- ❑ **Reuse across projects rarely happens by itself.**
 - A small amount of additional effort or funding is often enough to make software available outside the group that developed it
 - Significant barriers (e.g. intellectual property policies) may have to be removed or circumvented

- ❑ **A customized process is needed to**
 - Account for ESE's unique needs, culture, organizational structure and funding approaches
 - Engage and support the community
 - Avoid the common causes of failure in reuse initiatives
 - Leverage existing resources, infrastructure as well as lessons learned from other initiatives

- ❑ **Community opinion tended to bifurcate along two main themes**
 - **Mission-Critical**
 - Driven by launch schedules and a need for daily, highly reliable production, distribution and user services with operational promptness (e.g., SIPS, DAACs for standard products and high volume distribution)
 - **Mission-Success**
 - Driven by need for innovation in research in science, applications, or information systems by experimenting with differing products, approaches and mechanisms to adapt to new understandings (ESIP-2s, -3s, scientific analysis, etc)
- ❑ **Both communities are not satisfied with Status Quo and both agree that something needs to be done:**
 - Mission-Critical community is strongly in favor of an Improved Clone & Own (ICO) approach for itself
 - Mission-Success community favors the Open Source (OSS) approach for itself
 - Overall community opinion is strongly against a Product Line approach
 - Unlikely to realize theoretical benefits across a large and diverse enterprise like ESE

- ❑ Start with a simple process and engage all stakeholders in refining and evolving it
- ❑ Ensure that process is community-owned, non-prescriptive, scalable and practical
- ❑ Emphasize directly enabling reuse over infrastructure activities
- ❑ Leverage existing resources and infrastructure as well as lessons learned from other initiatives
- ❑ Use competition and peer review rather than blanket policies to help ensure that reuse always serves the ESE goals and does not become an end in itself
- ❑ Ensure that process can be tailored to different environments
- ❑ Remove barriers such as funding constraints, licensing issues, support concerns, security concerns, cultural issues, communication issues
- ❑ Establish incentives to encourage reuse and writing for reuse
- ❑ Facilitate sharing of knowledge and expertise
- ❑ Ensure that system and service providers maintain appropriate local control over their data system design, implementation and operation

Recommendation: Establish Two Working Groups

- One focused on the Improved Clone & Own approach in Mission-Critical environments
- One focused on the Open Source approach in Mission-Success environments

- ❑ Develop Draft Charters of Mission-Critical and Mission-Success Reuse Working Groups
- ❑ Gather Community Opinion on SEEDS Reuse Policy

- ❑ *Please indicate on the sign-up sheet whether you're interested in getting further involved in developing the working group charters and/or assisting in identifying and addressing policy issues*

Developing Draft Charters of Reuse Working Groups

❑ Reuse Working Groups

- Recommend initiatives and activities to the SEEDS program office based on assessment of upcoming mission needs, available assets and external reuse initiatives
- Recommend evaluation criteria for selection of reuse implementation projects
- Support incentive definition
- Develop and conduct outreach and education activities
- Assess policy change-related issues and support subsequent activities
- Contribute to support/enablement activities

❑ Issues to keep in mind for this session

- ❖ Different charters need to be created for the different environments (Mission-Critical and Mission-Success)
- ❖ Once formed, the Working Groups may edit and modify their charters
- ❖ The relationship between the SEEDS Integration Office and the Working Groups is not a tasking relationship
 - ❖ Each working group recommends initiatives and activities, and should have ownership of the strategies and activities used to meet the program goals
- ❖ Metrics need to be defined to measure the success of the Working Groups

- ❑ The charter is a document that outlines each Working Group's
 - Purpose
 - Scope
 - Membership
 - Goals & Metrics
 - Initial Steps

- ❑ **Sample Purpose from FGDC's Geospatial Applications & Services Working Group**
 - Further the goal of geospatial interoperability making a wealth of information available to benefit social, commercial, educational, sustainable future, land-use planning, agriculture, and crisis management decision-making.
 - Enable access to distributed, seamless data and enable merging of disparate data for user applications.
- ❑ **Purpose Common to both Reuse Working Groups**
 - *Enable/Facilitate Reuse through ...*
 - *Drive down the cost of system development?*
 - *Increase flexibility and responsiveness?*

Mission-Critical

- *Reduce cost?*
- *Improve reliability?*

Mission-Success

- *Broaden participation?*
- *Increase innovation?*

❑ Sample Scope from FGDC's Biological Data Working Group

- Focus on data standards pertaining to or descriptive of biological resources and ecological levels
- Help facilitate linkages and cross-walks between activities conducted by other Working Groups as they relate to biological data.

❑ Common Scope

- *focus on reuse of existing assets rather than reusability of newly developed assets?*
- *Not interfering with local control?*
- *define processes for establishing requirements or recommendations for making components reusable?*

Mission-Critical

- *Processes as applied to the Mission-Critical world?*

Mission-Success

- *Processes as applied to the Mission-Success world?*

What are the key responsibilities of Working Group members? And the time commitments expected from them?

❑ Sample Member Responsibilities from W3C's P3P Spec Working Group

- Participation requires that members become experts in P3P
- Members are expected to volunteer for assignments such as writing sections of specifications, creating examples, guidelines or test cases, reviewing implementations, and taking minutes
- Participation is expected to consume one day per week for each member's time
- In order to maintain good standing, members must consistently participate in 2 out of 3 working group meetings and complete working group assignments on time.

❑ Common Responsibilities

Mission-Critical

Mission-Success

What is the Criteria for Membership? Are there different membership levels?

❑ Sample Membership Criteria from W3C's P3P Spec Working Group

- To become a member, a representative of a W3C Member Organization must be nominated by their Advisor Committee Representative
- Membership is also open to invited experts from the community, selected by the chair in order to balance the technical experience of the group

❑ Common Criteria

- *CAN winners, ESE contractors, NASA personnel?*

Mission-Critical

- *Stricter criteria, more controlled membership?*

Mission-Success

- *Looser criteria, volunteer-based participation (with minimum level of commitment expected) ?*

□ Common Goals

- *Increase the percentage of projects applying reuse?*
- *Increase the number of qualified components available to system implementers?*
- *Decrease resources needed to locate and reuse existing components?*

Mission-Critical

Mission-Success

- ❑ Focus on performance-based metrics
- ❑ Each goal should be matched with a metric
- ❑ Examples
 - *Number of reusable components in repository?*
 - *Number of downloaded components/Number of users?*

- ❑ Focus on steps that need to be done first
 - *Set up repository?*
 - *Set documentation guidelines?*
 - *Define criteria for selecting reusable components?*

- ❑ Extract first steps from previously-identified goals

Gathering Community Input on SEEDS Reuse Policy

- Licensing and contractual issues were identified as potential barriers to the recommended ESE reuse process

What are the community's concerns regarding these issues?

□ 3 Themes for Questions about Policy

➤ Current Policies

- Can a reuse activity succeed given the current policies?
- Can we identify which policies may impede reuse?
- What policy changes can be recommended?

➤ Perspective of External Organizations

- Could policies of centers, universities or commercial organizations interfere with a successful reuse activity?
- What are the licensing options available, and how can they possibly impact external organizations' participation in the process?

➤ Recommendations

- What types of policies are needed to ensure the process is a success?
- What types of changes in the current contract wording can be recommended to facilitate the implementation of the recommended process?
- What is the best policy that will enable reuse? And what is the best way to state that policy?

□ Key Questions

- Can a reuse activity succeed given the current policies?
- Can we identify which policies may impede reuse?
- What policy changes can be recommended?

□ Initial Discussion Points

- External Release of NASA Software NPG
 - We are interested in learning about your experiences with releasing software (either to other NASA centers or outside of NASA)
- CAN Data Rights and Related Issues
 - We are interested in hearing about whether the data rights outlined in the REASoN CAN may restrict your participation
- NASA Standard Contracts and Agreements
 - We are interested in learning about your experiences with the current wording of contracts and about any recommendations you think will encourage reuse



□ External Release of NASA Software

- Outlines procedures and guidelines for the external release of NASA software
- Used even for transferring software from one NASA center to another
- Scope
 - Software created by or for NASA from work performed by employees of NASA and NASA contractors, either solely or jointly among multiple parties, and from work performed by NASA employees working with non-federal parties with or without an underlying agreement
 - Software definition used
 - Means computer programs, executables, source code and object code
 - Does not include computer databases or software documentation
 - Design details, algorithms, processes, flow charts, formulae can only be released if the associated software has been approved for release
- Release scope: Public domain (OSS), US & Foreign, US only, US Government only, NASA only
- Parties involved in the process
 - Commercial Technology Division in the Office of Aerospace Technology at Headquarters
 - NASA Inspector General, NASA Principal Center for Information Technology Security (PCITS)
 - Centers' Patent or Intellectual Property Counsels
 - Centers' Commercial Technology Divisions, Centers' Software Release Authority (or Group)
 - Centers' IT Security Managers, Centers' Export Administrators, Centers' Procurement Offices

□ *We are interested in learning about your experiences with releasing software (either to other NASA centers or outside of NASA)*

❑ CAN Data Rights and Related Issues (Appendix G)

➤ Information Systems software is treated as “Data First Produced by Recipient”

- Software that comprises any part of, access to, or management of data in the data system of the REASoN project, or tools that access, manipulate, or analyze Scientific Computer Data Base
- If the Recipient intends to **commercialize** data first produced by the recipient
 - It shall be responsible to ensure such data is appropriately marked with a suitable notice
 - The Data will be maintained **in confidence for a period of two years after completion** of this agreement and be disclosed and used by the Government and its contractors (under suitable protective conditions) **only for use as a tool for Government research** by or on behalf of the Government during that period
 - In order that the Government and its contractors may exercise the right to use such Data for the purposes designated above, NASA, upon request to the Recipient, shall have the right to review and request delivery of Data first produced by Recipient
- If the Recipient determines it is **not going to commercialize** (or otherwise make available to the user community such as freeware) such data, or if the Recipient fails to provide written documentation indicating its intent to commercialize the data
 - The Recipient agrees that all such data can be made **available without restriction** as to its disclosure, use or duplication.
 - Alternatively, at NASA’s option, NASA may require the recipient to assign any copyright to such data to NASA or its designee.

❑ *We are interested in hearing about whether the Data rights outlined in the REASoN CAN may restrict your participation*



❑ NASA Standard Contracts and Agreements

- Most standard NASA contracts and agreements allow for software to be reused on other government projects
- The government always retains the right to use the code

- ❑ *We are interested in learning about your experiences with the current wording of contracts and about any recommendations you think will encourage reuse*

□ Key Questions

- Could Center, University or Commercial Organization policies interfere with a successful reuse activity?
- What are the licensing options available, and how can they possibly impact external organizations' participation in the process?

- *We are interested in hearing from you about your organizations' policies, and whether these may impact your participation in a reuse activity*

□ Key Questions

- What types of policies are needed to ensure the process is a success?
- What types of changes in the current contract wording can be recommended to facilitate the implementation of the recommended process?
- What is the best policy that will enable reuse? And what is the best way to state that policy?

□ Initial Discussion Points

- Issues to consider when selecting/creating licenses
- From community requirements to reuse policy elements

❑ Creation of derived works

- Do you want people to be able to make modifications private or not?
- Do you want contributors to document and be responsible for these modifications (including support)?
- Do you want contributors to notify community (or control authority) of major modifications/enhancements?
- How do you want to assign IP rights?
- Do you want the original copyright holder to have any special privileges over modifications?

❑ Redistribution

- Do you want some people to be able to buy commercial-licensed version of your program that is not open source?
- What scope of redistribution do you want?

❑ Implications when merging or linking software with other proprietary software

- Do you want to allow someone to merge your program with other proprietary software?

❑ Degree to which the license incentivizes future developers to contribute

Policy: From Community Environment Requirements to Reuse Policy

SEEDS FOR-FAITH

Mission-Critical **Improved Clone & Own**

- driven by cost and schedule
- Risk averse
- Trusted development
- Expert development
- Funding sources: more from government contracts

Mission-Success **Open Source**

- Driven by innovation
- Collaborative development
- Merger of inputs to coherency
- Greater service commercialization potential

Characteristics and
Needs of Community



Building Blocks of
Reuse Approach



Elements of
Supporting Policy

- Moderator likely not needed
- Controlled trusted library of source code, documentation, test suites
- Relatively smaller community
- Support from community experts

- Open library
- Control authority needed
- Larger community

- Proposal winners have Copy access by password
- Everyone else has Read access
- Bidders must examine library

- Everyone has Read and Copy access

Backup Material

❑ GNU Public License (GPL)

- E.g., Kerberos
- All changes have to be returned to the community with source code
- License only inquires open inspection if software is redistributed
- New product and changes are automatically covered under the GPL

❑ Lesser or Library GNU Public License (LGPL)

- E.g., OpenOffice
- For situations when developers might only link to libraries and not modify them
- Permits use of library in proprietary programs
- Derivative works need not be distributed under this license

❑ Berkley Software Distribution License (BSD)/MIT License

- E.g., Apache
- Only requirement is to publish the original copyright notice in the source code or in the product documentation
- Here's the code, do what you like with it, just give us credit when you try and sell it

❑ Mozilla Public Licence (MPL)

- E.g., Mozilla
- Allows developers to create a larger work by combining source code from an MPL licensed product with their own source code without requiring them to release the new product under the MPL (provided the new work is in completely separate files)
- Mandates that the company or individual contributing code back to the project release any and all claims to patent rights that may be exposed by the code
- Netscape Public License (NPL) gives Netscape the privilege of re-licensing modifications that other people made to their software (ie, they can take your modifications and make them private)

❑ Artistic License

- E.g., Perl
- Allows creation of derivative products without disclosing source code of the modifications as long as the resultant work is used internally within an organization
- Prohibits sale of software but allows an aggregate software distribution of more than one program to be sold (people often bundle a hello.c program and they get away with it)

Common Open Source Licenses

License	Can be mixed with non-free software	Requires publication of source code of any modifications or derived works	Can be re-licensed by anyone	Contains special privileges for the original copyright holder over contributed modifications
GPL	No	Yes	No	No
LGPL	Yes	Yes but code that only links to the original code is not subject to this requirement	No	No
BSD	Yes	No	No	No
MPL	Yes	Yes, but additions contained in completely separate files are not subject to this requirement	No	No
NPL	Yes	Yes	No	Yes
Artistic	Yes	Yes unless modifications are meant for internal use within an organization	Yes	No

❑ Microsoft's Shared Source

- Allows access to the source code for non-commercial purposes
- No modifications or derived works are allowed

❑ Sun's Community Source License

- Premise:
 - Community of common interests centered around an infrastructure which is provided by a particular organization (the developing organization)
 - Other organizations having an interest in building businesses around the infrastructure may join the community
 - The original source base is fully the responsibility of the developing organization which can move that base forward
- The community comprises of only those who have agreed to the license
- Policy on modifications
 - Error corrections must be given back to the community
 - Proprietary modifications and extensions including performance improvements are allowed
 - Community members may contribute shared modifications to the community, granting rights to the community to use these modifications, including intellectual property, specifications and test suites.

❑ UK OS Policy

- Consider OSS solutions alongside proprietary ones in IT procurements
- Only use products for interoperability that support open standards and specifications in all future IT developments
- Consider obtaining full rights to bespoke software code or customizations of COTS software procured wherever this achieves best value for money
- Explore further the possibilities of using OSS as the default exploitation route for Government funded R&D software

❑ Grid for UK Particle Physics OS Policy

- All software developed is Open Source
- OS software to be licensed through the EDU DataGrid or use a license based on the EU DataGrid Software License
 - You are under no obligation to provide anyone with any bug fixes, patches, upgrades or other modifications, enhancements or derivatives of the features, functionality or performance of this software that you may develop. However, if you publish or distribute your modifications, enhancements or derivative works without contemporaneously requiring users to enter into a separate written license agreement, then you are deemed to have granted participants in the EU DataGrid a worldwide, non-exclusive, royalty-free, perpetual license to install, use, reproduce, display, modify, redistribute and sub-license your modifications, enhancements or derivative works, whether in binary or source code form, under the license conditions stated in this list of conditions.

❑ University of WA IT Policy

- Actively encourage the exploration of OS solutions in all areas of application
- Provide, for software intended for distribution as OS, a general waiver of the current requirement of university IP policy...
- Ensure that university is not imposing any requirements or expectations that would require students to necessarily make use of proprietary software
- Investigate the impediments to wider deployment of OS and how to overcome any difficulties associated with living in a mixed environment

Definitions

❑ Free redistribution

- License shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources

❑ Source code

- License must allow distribution of software in source code as well as compiled form
- License may include well-publicized means of obtaining the source code

❑ Derived works

- License must allow modifications and derived works.
- License may allow modifications to be private (BSD) or not (GPL)

❑ Integrity of author's code

- License may restrict source-code from being distributed in modified form only if the license allows the distribution of patch-files with the source code for the purpose of modifying the program at build time
- License must explicitly permit distribution of software built from modified source code
- License may require derived works to carry a different name or version number from original software

❑ Distribution

- Rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties

❑ Restriction on other software

- License must not place restrictions on other software that is distributed along with the licensed software

❑ Discrimination

- License shouldn't discriminate against persons or group
- License shouldn't discriminate against fields of endeavor

❑ No warranties

❑ Reuse

- Taking a functionality used in (or provided by) one system or mission and employing it in another system or mission
- This functional capability can be in the form of code, or it can be design “artifacts” (e.g. architectures, software designs, ICDs, test plans, etc)
- Broad definition for this study encompasses any means that avoids rebuilding a capability

❑ Reference Architecture

- A generic architecture for use in particular domain (e.g. Earth science)
- Used as a reference when developing a specific architecture
- Provides a common reference to promote component reuse, reduce integration costs and promote interoperability
- Focus is on enabling application (domain-specific vs. infrastructure) software reuse and application system openness

❑ Clone & Own

- Copying code and associated artifacts of one system for use in another system, where they may be independently modified and maintained

❑ Improved Clone & Own

- Extending Clone & Own practices to enable developers to identify existing assets easily, subsequently copy those assets, modify and integrate them more easily as needed for use in new systems

❑ Open Source

- Engaging developers across missions to collaboratively develop and update selected components or systems stored in a central repository

❑ Mission-Critical Activities

- Activities driven by launch schedules and a need for daily, highly reliable production, distribution and user services with operational promptness (e.g., SIPS, DAACs for standard products and high volume distribution)

❑ Mission-Success Activities

- Activities driven more by need for innovation in research in science, applications, or information systems by experimenting with differing products, approaches, and mechanisms to adapt to new understandings (ESIP-2s, -3s, scientific analysis, etc)

❑ Reuse Initiative

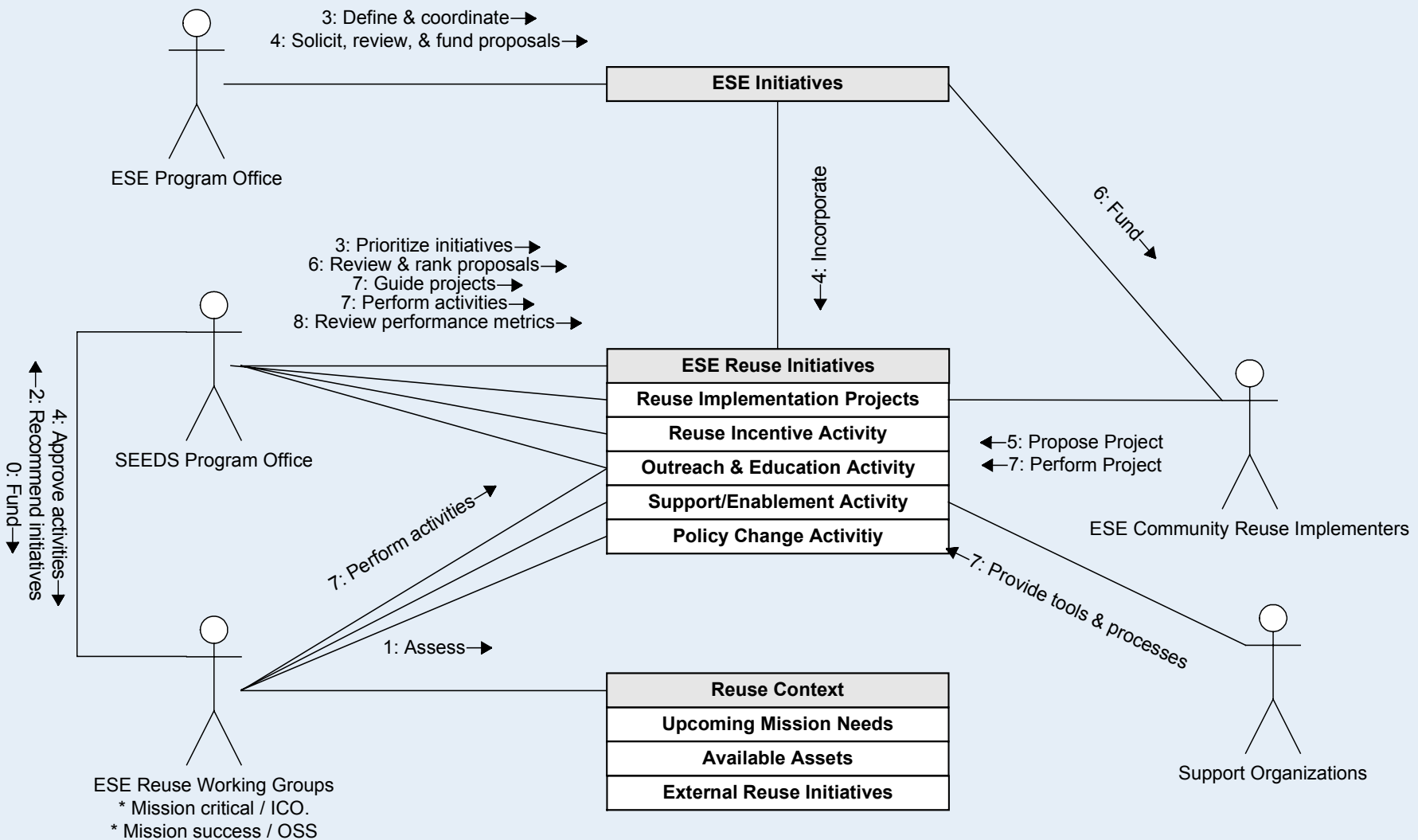
- Process aimed at achieving a specific reuse goal (such as making reusable assets available or better publicizing existing assets)
- Can be accomplished through a variety of activities

❑ Reuse Project/Activity

- Effort aimed at supporting the execution of a reuse initiative
- Project/Activity categories
 - Reuse implementation projects
 - Reuse incentive activity
 - Outreach and education activity
 - support/enabement activity
 - Policy change activity

Next Steps

1. **Define and initiate community-owned, non-prescriptive, scalable and practical processes to implement community recommendations**
2. **Establish two working groups**
 - One focused on the Improved Clone & Own approach in Mission-Critical environments
 - One focused on the Open Source approach in Mission-Success environments
3. **Establish a separate body such as a Reuse Integration Office**
4. **Complement the reuse effort with an effective technology development and technology infusion effort to bring in new and increased functionality**



❑ Reuse Implementation Projects

- Efforts that directly result in the publication or use of a reusable component
- Examples:
 - Component generalization and documentation
 - Reusable component integration pilots

❑ Reuse Incentive Activities

- Awards and structural changes that directly & indirectly encourage reuse
- Examples:
 - Small competitive or lottery-type monetary awards for submitting components to a repository, authoring a popular component, etc.
 - Clauses in technology program grants/contracts that permit using funds saved through reuse for further R&D

❑ Outreach and Education Activities

- Efforts that increase community awareness and understanding of benefits, best practices, tools, available components, etc.
- Examples:
 - Conference workshops
 - Informational Web sites and newsletters

❑ Support/Enablement Activities

- Efforts that provide tools and mechanisms to enable reuse
- Examples:
 - Implement (not develop) asset catalogs and open source community Websites
 - Disseminate license templates that promote reuse

❑ Policy Change Activities

- Efforts to reduce policy barriers to reuse
- Examples:
 - Revise open source software policies

- ❑ **Define process or set of processes which ESE can use to evaluate and make decisions on establishing requirements or recommendations for making components reusable**
 - Identify process for adding proposed components to catalog of components
 - Identify guidelines to help communities evaluate whether to add software to the catalog of components
 - Provide checklist of items that must be provided to make assets reusable
 - Set policies, targets and metrics for level of reuse
 - Review needs of upcoming missions, compare these needs against the current catalog of assets and external initiatives to identify gaps
- ❑ **Support incentive definition**
 - Identify, assess, & recommend incentives
- ❑ **Conduct education & outreach**
 - Conduct workshops/forums to share lessons learned, advertise ideas, solicit input from the community, etc
 - Establish reuse website, newsletter, etc
 - Provide interface to external organizations and other federal/international agencies/institutions to track their activities and represent ESE initiatives in their efforts
- ❑ **Conduct support/enablement activities**
 - Document requirements and guidelines
 - Select standard for architecture expression to enable communication about available resources
 - Identify and prioritize synergistic activities
 - Define needed infrastructure for reuse
 - Identify areas of high reuse potential
 - Provide consulting services to the community
 - Support groups in their open source planning phases
 - Support groups in making their software reusable
 - Work with groups throughout implementation/integration phases
 - Setup and maintain a catalog of reusable components
 - Examine open source issues, provide templates, etc
 - Qualify candidate initiatives against SEEDS program goals
- ❑ **Support policy changes activities**
 - Identify & recommend needed policy changes
 - Address intellectual property issues